

Amendments to the Claims:

The following listing provides the amended claims with deleted material crossed out and new material underlined to show the changes made.

Listing of Claims:

1. (Currently Amended) A method comprising:

for an event to be logged that has not yet been logged within an application,

~~creating, for an event to be logged that has not yet been logged within an application,~~
an event object, said event object occupying a memory space ~~and executing that~~ is independent of
said application; and

logging within said event object the start time, end time and information regarding the
event.

2. (Currently Amended) A method according to claim 1 further comprising:

~~checking, for each event identified by the application,~~ whether event logging has been turned
on for the event.

3. (Previously Presented) A method according to claim 2, wherein said creating and said
logging are performed for each event having event logging turned on, wherein a plurality of event

objects are created and logged for a plurality of events.

4. (Currently Amended) A method according to claim 3 further comprising:

analyzing of said event objects after event logging is turned off.

5. (Currently Amended) A method according to claim 4, wherein analyzing includes:

allowing user definition of the hierarchical levels of granularity of said events whose event objects are to be analyzed; and

allowing user definition of contexts for differentiating repeated occurrences of events deemed identical by nature of their hierarchical position.

6. (Currently Amended) A method according to claim 5, wherein analyzing further includes:

grouping events into their hierarchical subgroups; and

grouping events by their context, if any are defined.

7. (Currently Amended) A method according to claim 6, wherein analyzing comprises:

traversing through the hierarchy of subgroups until the subgroup of finest granularity is traversed;

subdividing said events into further subgroups;

computing statistics for each subgroup while traversing; and

displaying said statistics.

8. (Currently Amended) A method according to claim 7, wherein if said subgroup of finest granularity has been traversed, then:

aggregating events deemed identical by virtue of their hierarchical position into an aggregate;

computing statistics for each aggregate; and

displaying said statistics for each said aggregate.

9. (Currently Amended) A method according to claim 7, wherein said analyzing includes:

aggregating events deemed identical by virtue of their context into an aggregate;

computing statistics for each aggregate; and

displaying said statistics for each said aggregate.

10. (Previously Presented) A system comprising:

a foundational layer upon which applications are built and executed; and

an event logging mechanism executing independently of said applications, said mechanism for generating an event log for any of said applications, without referencing any event logs of said applications.

11. (Currently Amended) A system according to claim 10, wherein said event logging mechanism logs start time, end time and other event information into the event object for each event to be logged.

12. (Currently Amended) A system according to claim 10, wherein said foundational layer is an operating system.

13. (Currently Amended) A system according to claim 10, wherein said foundational layer is a programmable framework.

14. (Currently Amended) A system according to claim 11, wherein said event logging mechanism can be turned on and then off from beyond the execution space of said applications within said foundational layer, said turning on and off separate for each event.

15. (Currently Amended) A system according to claim 10, wherein said event logging mechanism can be turned on and turned off and configured using a browser application.

16. (Currently Amended) A system according to claim 15, wherein said event logging mechanism generates a plurality of event objects and is configured to analyze said event objects and

present to said browser application the results thereof.

17. (Currently Amended) A system according to claim 16, wherein said event logging mechanism is configured to analyze said event objects based upon hierarchical and contextual grouping.

18. (Currently Amended) A system according to claim 16, wherein said event logging mechanism is configured to aggregate said event objects deemed identical based upon at least one of hierarchical and contextual grouping.

19. (Currently Amended) An article comprising a computer readable medium having instructions stored thereon which when executed causes:

for each event in a plurality of events to be logged that has not yet been logged within an application,

~~creating, for an event to be logged that has not yet been logged within an application~~
an event object, said event object occupying a memory space ~~and executing~~that is independent of said application; and

logging within said event object the start time, end time and information regarding the event.

20. (Previously Presented) An article according to claim 19 having further

instructions stored thereon which when executed causes:

analyzing of said event objects according to hierarchical and contextual grouping.

21. (Currently Amended) An apparatus comprising:

means for creating, for ~~every~~an event to be logged that has not yet been logged within an application, an event object, said event object occupying a memory space ~~and executing~~that is independent of said application; and

means for logging within said event object the start time, end time and information regarding the event.

22. (Currently Amended) An apparatus according to claim 21 further comprising:

means for analyzing of said event objects according to hierarchical and contextual grouping.

23. (Currently Amended) A system comprising:

a foundational layer upon which applications are built and executed;

a first application for executing on said foundational layer,

a second application for execution on said foundational layer,

~~a third~~an event-logging ~~application~~mechanism for execution on said foundational layer, for

functioning interoperably with but separately from said first and second applications, and for generating an event log for ~~each~~either of said first and second applications, wherein at least one of said first and second applications does not generate an event log.

24. (Currently Amended) A system according to claim 23, wherein said generating an event log comprises storing, for each event to be logged, a temporal attribute of ~~the~~an event in the event object associated with the event.

25. (Currently Amended) A system according to claim 23, wherein said ~~third~~-event-logging mechanism comprises analyzing of said event log according to hierarchical and contextual grouping.

26. (New) A system according to claim 23 further comprising a first area of memory allocated to the first application, a second area of memory allocated to the second application and a third area of memory allocated to the event logging mechanism, wherein said third area of memory is separate from the areas of memory allocated to the first and second applications.

27. (New) A system according to claim 26 further comprising an enable/disable state for each event identified by the application wherein the disable state precludes any system from creating an event log.

28. (New) A system according to claim 26, wherein generating an event log is performed for each event having event logging enabled.

29. (New) The system according to claim 23, wherein the foundational layer is a development framework upon which applications are built.

30. (New) A event logging method comprising:

for each of a plurality of events that need to be logged but have not yet been logged within a plurality of applications,

creating an event object,

storing said event object in a first memory space that is uniquely allocated for the event logging method, said first memory space separate from a second memory space allocated for the plurality of applications; and

logging within said event object the start time, end time and information regarding the event.

31. (New) A method according to claim 30 further comprising creating, for the event object, an enabled/disabled status wherein the disabled status disables all logging for the event within a system that includes the plurality of applications.

32. (New) A method according to claim 30 further comprising checking, for each event identified by an application within the plurality of applications, whether event logging has been enabled.

33. (New) A method according to claim 32 further comprising analyzing of said event objects after event logging is disabled.

34. (New) A method according to claim 30, wherein the memory space occupied by the event log is within memory space that has been allocated solely to the event logging mechanism.

35. (New) A method according to claim 30, wherein the events that are logged by the event logging mechanism have not been previously logged by any other application.

36. (New) A method according to claim 30, wherein information placed in the event log is first logged by the event logging mechanism.

37. (New) A method according to claim 30 further comprising an enable/disable state wherein the disable state precludes any system from creating an event log.

38. (New) A method according to claim 30, wherein the foundational layer is not part of the operating system layer.